

**Anonymous reviewer 1, from 03 Nov 2021**

*In "Correlation dispersion as a measure to better estimate uncertainty of remotely sensed glacier displacements", Altena et al. present an efficient and straightforward method for quantifying spatially-varying precision associated with displacement maps derived from pattern matching (pixel tracking) techniques. These techniques have become ubiquitous in glaciology and are the foundation for most contemporary velocity fields derived from remote sensing data. Here, Altena et al. utilize the topology of local cross-correlation surfaces to estimate the offset precision, which plugs in nicely to most existing workflows and should be robust for most datasets used in glaciology. Importantly, they show how these uncertainties can be linked to distinct classes of surface features, such as crevasses or shear patterns.*

*Overall, this manuscript represents an important and long overdue contribution to the field and to any researcher using remotely-sensed velocity fields. In particular, any work on data assimilation or inverse modeling absolutely needs to properly account for spatially-varying uncertainties. My comments are mostly minor/moderate and are mostly concerned with improving the presentation of the results and placing the results within the context of prior techniques and uncertainty measures.*

Thank you for your kind words.

**Summary comments**

*1) Most of the uncertainty metrics presented in the main text are those quantifying correlation surface orientation and degree of asymmetry. However, as a user of velocity fields reading a manuscript on uncertainty quantification, I did not actually see a map of velocity uncertainties for any of the study sites! I believe these maps should be front-and-center in the main text so the readers can immediately obtain an intuition on the spatial distribution of uncertainties and their relation to glacier flow speed. In my opinion, the other uncertainty metrics (e.g., correlation peak orientation and elongation) are interesting auxiliary results which should exist to enhance the presentation of the uncertainties themselves, since the latter are what most people working on data assimilation will be interested in.*

We follow this line of thinking, and the reviewer is right about this issue. We have now included the along and across flow estimates in the main figure of the results.

*2) When showing the maps of uncertainties, it would also be very useful to show how uncertainties change for different window sizes. For users wanting to adopt the methods in this manuscript, a demonstration of uncertainties for common window sizes (64x64, 128x128, or adaptive window sizes) would go a long way towards bringing awareness to window size effects on both velocity field noise levels and uncertainties.*

We think this will seriously deviate from the message, literature is full of such empirical studies. We agree with the reviewer that window size does matter (see also our response below to reviewer 2 about approaching this as an optical flow problem). However, this is only one part of the equation, as the image content contributes as well. Hence, the results will depend on the scene and window-size used.

Nor do we want to restrict the interested reader from such exploration, therefore the processing code is available. Hence, if such subjects are of interest to the reader, the provided scripts can be used for exploration on this matter. In sum, we prefer therefore to not implement the suggestion of the referee for the sake of focus of the paper.

The provided second point of the feedback of the reviewer is well appreciated, but though interesting, the inclusion of a least-squares matching implementation (which is required for an adaptive window approach), will also deviate from the analysis given here.

*3) Pattern matching/pixel tracking has a long history in fluid mechanics (PIV, as the authors mentioned) and geosciences. For the task of deriving velocity fields over terrestrial ice, these methods have been used extensively for over*

20 years, and several of the largest projects (e.g., MEaSUREs) also provide maps of velocity errors for their velocity products. While a brief discussion on previous uncertainty quantification methods was included in the introduction, there needs to be additional discussion and comparison of the uncertainties for methods that also go beyond the homoscedastic assumption. For example, the method used by Joughin, 2002, "Ice sheet velocity mapping..." computes the statistical offset variance for a local window centered on a given pixel. Thus, the scatter in the estimated correlation peak represents an aggregate of the different noise factors for the local window. My intuition is that the method used here (fitting the correlation surface) is a more robust approach, especially for smaller window sizes, but without a comparison of uncertainty maps for the different methods, it's hard to know for sure. Thus, related to comment (1), my suggestion is to include an uncertainty map (at least for Sermeq Kujalleq) and compare that map with a similar map from the Greenland Ice Sheet Mapping Project (GIMP).

We understand the point made by the reviewer, but if we want to keep close to the message given by the manuscript, this does look like a strong deviation thereof. The approach presented here is an extra feature within the processing chain, which has been introduced earlier (see comments later given by this reviewer). It gives extra handholds for more advanced post-processing, such post-processing procedures that are here asked for.

The mentioned post-processing procedure of Joughin (but also many others) has a strong assumption on the homogeneous consistency in flow direction and its velocity gradient. Though usefull for large outlet glaciers, it is a procedure that is further down the processing chain, from velocity pairs to mosaics. Nevertheless, we do see the benefit of such a comparison here in the response, as it might give insights on the total processing chain. Therefore, we have replicated the workflow as presented in (Joughin, 2002) and GIMP.

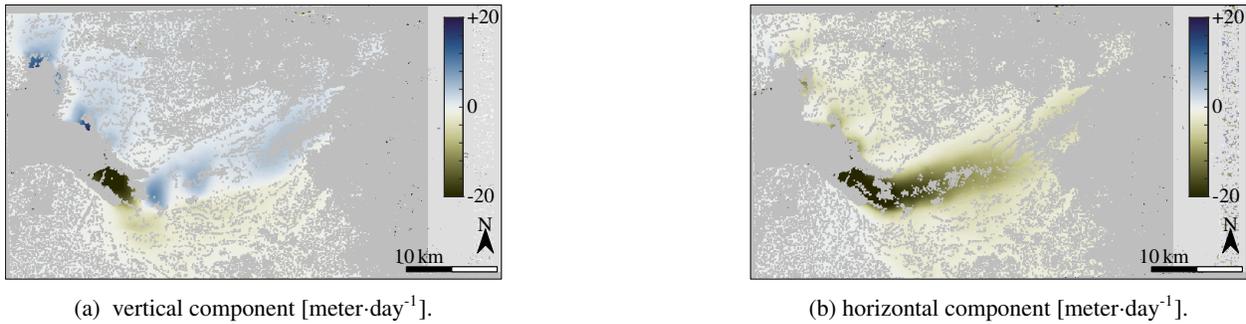


Figure 1: Raw velocity estimates.

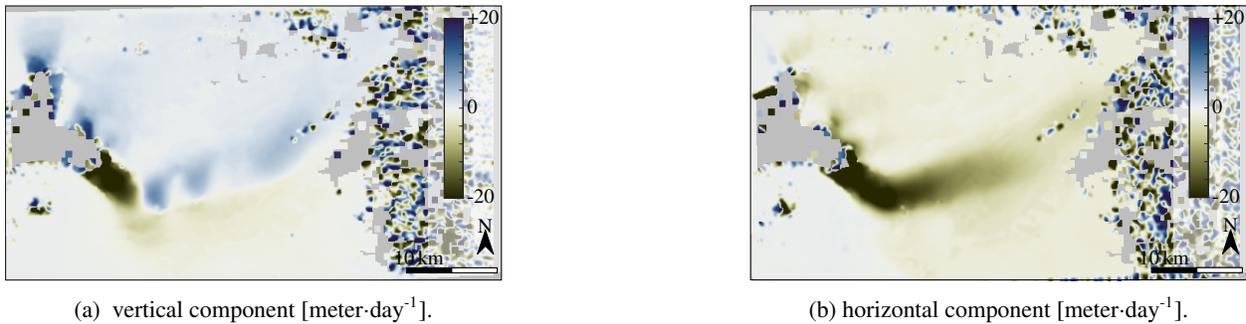


Figure 2: Culled, filled and smoothed velocity estimates, following the workflow of (Joughin 2002) .

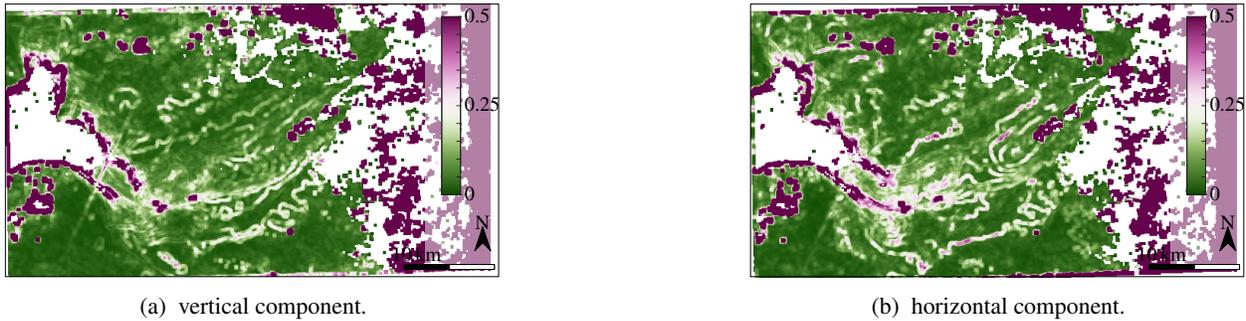


Figure 3: Variance estimate following the workflow of (Joughin 2002), and as is given by GIMP .

Apart from being scale dependent (see provided code for specific parameter implementations), in this implementation the variance estimates seems most sensitive to shear zones. Since our method is based on the image content, it does highlighted the crevassed regions north of the outlet.

4) Related to comment (3), estimation of pattern matching uncertainty by quantifying the topology of the cross-correlation surface has been around for a while. In the field of InSAR, software packages like the Repeat Orbit Interferometry Package (ROI\_PAC) and InSAR Scientific Computing Environment (ISCE) estimate the curvature of the oversampled cross-correlation surface as an uncertainty proxy, similar to what's done here (see Rosen et al., 2004, "Updated repeat orbit interferometry package released" and the appendix of Casu et al., 2011, "Deformation Time-Series Generation ..."). At a minimum, these should be cited in the manuscript.

We would like to thank the reviewer for pointing this out. We have been aware of AmpCorr as it has been a subroutine used by many other processing pipelines, but the mentioning of this curvature estimate has not been mentioned explicitly.

The backbone of this package is described in the thesis of Rosen, but it distribution/accesability is less optimal. Unfortunately, the provided references do not specify how the hessian is estimated. Luckily the fortran code of AmpCorr is available, hence we have tried to translate this to Python code, so we were able to compare both methods.

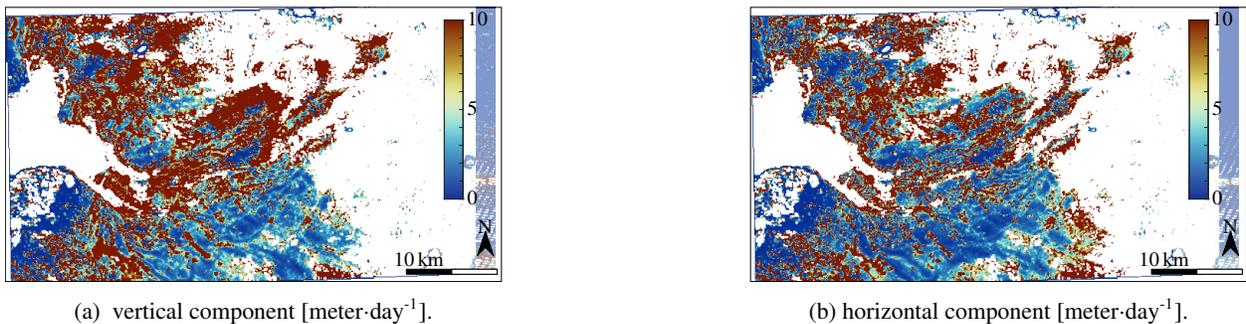


Figure 4: Curvature estimation as done in a similar fashion as ROI\_PAC.

There is a strong agreement between both methods, crevassed regions in the center of the scene have similar patterns. However, differences are present as well. Most pronouced is the stronger differentiation between sharp or wide peaks.

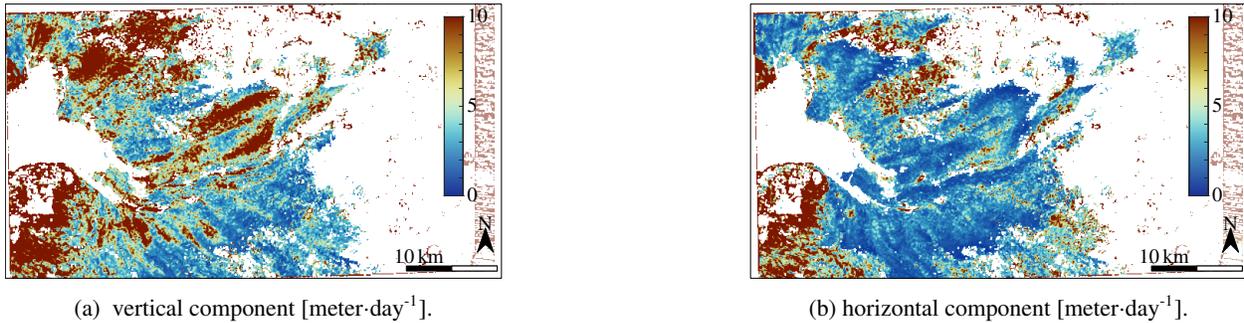


Figure 5: Dispersion estimation as proposed by (Altena et al.) .

This can be due to a smaller amount of neighboring pixels that is used for the estimation. For the Gaussian model takes a larger subset surrounding the peak. Another pattern, is the polar opposite of the estimates in the South Western side, where stable terrain is situated. Here there is a lack of structure, causing either very elongated or very narrow lobes to be estimated.

**Line-by-line Comments:**

- *Line 61: A comparison of most of these methods has been done in the field of PIV, and at least one citation should be included (e.g., see Xue et al., 2014, "Particle image velocimetry correlation signal-to-noise ratio metrics...").*

We have included this suggestion, and we agree that many algorithms and methods seems transferable. However caution should be taken, since this field is based on laboratory measurements where a controlled environment is the usual setup. Hence, the PIV data has a clear distinction between "data-points" carrying displacement information and "background noise". For example the work by (Wieneke, 2015) is based on knowledge of the particle seeding diameter, for image matching in the wild this is less clear. The controlled environment of PIV setups makes it also easier to derive proxies or propagate errors through the whole pipeline, as in (Bhattacharya et al. 2017). Hence, the large body of work in this domain is interesting but separating particle image diameters, seed densities, displacements, and velocity gradients to come to an error budget might not be practical or easily transferable to the case presented here.

- *Section 3.1: Is any oversampling of the cross-correlation surface performed prior to fitting a 2D Gaussian? Oversampling is a common step in pattern matching and can mitigate the effects of "pixel locking" (see autoRIFT paper, Lei et al., 2021, Remote Sensing; a Gaussian pyramid upsampling scheme is used to reduce pixel locking). Additionally, it would likely provide more data points for the 2D Gaussian fit. Without oversampling, wouldn't there be situations where the cross-correlation surface is highly concentrated at a single location (i.e., a high SNR case), in which case the 2D Gaussian fit would be poorly conditioned.*

Since spatial domain image matching is performed, no oversampling is applied. Oversampling is a strategy to suppress peak-locking when frequency domain methods are applied (see comment for line 257).

Looking at the results at the Figure 4 and 5 above, the indication given by the reviewer seems to go towards the opposite direction in this case. It seems the parameter estimation through an Hessian, using 5, 7 or 9 data points, makes the estimation too optimistic or exaggerate the lobe. Using 25 data points does seem to benefit the estimation, but if the window size shrinks or the image content changes to speckles, this might not be the case. To a degree interpolation is present in the procedure, since a weighting function is optionally implemented in the code.

*Also, what is a typical zoom window size (centered around the peak) for fitting the Gaussian? I assume the entire cross-correlation surface is not used for the fit. If the zoom window size encompasses multiple peaks (primary + secondary peaks), how is that situation handled in the processing chain?*

This crucial information was not included in the text, but now: “The direct neighborhood is used here (radius=1) when the peak next to the border, otherwise a two pixel radius is used.” is included.

In theory the second peak might be present in this outershell, and would corrupt the estimation. One strategy is a distance weighting, as mentioned above. A second strategy is to account for the corruption through data-snooping, but we did not experienced such defects, and this is not implemented, partly because the exponential function corrupts the noise distribution.

*- Line 119: I think care should be taken when referring to general covariance matrices. In general, off-diagonal elements describe dependences between variables. The temporal/spatial relational dependencies mentioned on this line are for a completely different set of variables.*

We think it should be mentioned here, to give an outlook where this can go to. Seen from a bundle block adjustment perspective, it is logical that if a network of imagery are matched, the dependencies should be included. A sparse matrix is computationally more friendly, but formulations with off-diagonal elements might make it easier to identify outliers and biases.

*- Figure 2: This figure could likely go into Appendix B since, by itself, it doesn't add too much to the discussion.*  
Has been moved to the appendix.

*- Line 152: It's a bit incomplete to say that co-registration is not applied to the image pair beforehand. Nominally, users will use the image metadata to approximately co-register the images in order to reduce the need for large search windows. Are the authors referring to refinement of image registration over stable ground? If so, that should be stated more clearly.*

Registration of Sentinel-2 is currently done through its sensor recordings, though current data does use the Ground Reference Image (GRI). Legacy imagery does fluctuate within an absolute registration error bound of 10 meter (Kääb et al. 2016). We have now included the mentioning of orthorectification, though the Level-1A data is not available. The suggestion by the reviewer is a bit confusing, as we follow the terminology of (Nuth & Kääb), since neither Sentinel-2 scenes are ground truth, and a relative off-set/ mapping function is not estimated.

*- Line 163-164: What does "extensive flow" mean? Extensive extensional strain?.*  
Has been adjusted.

*- Figure 3/4: I suggest moving the Radon crevasse orientation in Figure 3 to Figure 4 to better compare with the correlation peak orientation..*  
The figure has been moved.

*- Figure 8 caption: Please also include that these results are for Sermeq Kujallec..*  
The glacier name is now included.

*- Line 212: Doesn't strong shear flow generally result in crevasse formation? It seems a little odd to categorize these features into two distinct classes.*  
This is true, but crevasses are transported, where they still can have an effect on the correlation spread.

*- Line 221-222: If I'm not mistaken, pixel/peak locking is a consequence of estimating the center of mass of a few discrete pixels. Methods that fit the correlation surface with a model (as is done here) should thus avoid those issues, right? Can't peak locking also be mitigated by oversampling of the cross-correlation surface? (see my comment above*

*about oversampling of the cross-correlation surface).*

Peak-locking issues arise distinctively when frequency based methods are used and this issue is raised as this bias is systematically favoring lower error statistics. The localization of this peak location is not analyzed here, neither does the dispersion estimation compensate for such biases.

Oversampling might work, another method to compensate for this peak-locking bias is shifting by .5 pixels.

*- Line 224: Please add a few words on what least squares matching is. What do "intensities" refer to here?*

We have limited ourselves to a brief description of template matching within a least-squares framework. Hence, the paper by (Förstner, 1987) roots to this concept and connects to the work by (Maas et al.). We find it out of scope for this paper to describe template matching with an affine model, which is typically done in a least squares framework as well, and is typically known as least squares matching.

*- Line 232: descriptor change to descriptor*

Corrected.

*- Line 234-238: These sentences are a bit confusing to me. I don't quite understand how the sub-pixel displacements influence the correlation score. Is it because a correlation peak's energy becomes evenly distributed across multiple pixels in the cross-correlation surface? If so, this seems similar to the pixel locking effect and could perhaps be mitigated by oversampling of the cross-correlation surface (see my earlier comment).*

Here we have used spatial domain matching, hence we do not expect peak-locking to be very much pronounced. Peak-locking occurs due to sharp sidelobes, as pure displacement through frequency methods resemble a sinc-function. Thus correlation scores next to the maximum can have negative values. This will influence the location estimate if a gaussian or polynomial model is used. Hence, we do not think this pattern is peak locking, and this is only an observation showing the absolute correlation value is influenced by the underlying sampling.

*- Line 239: precision change to precision*

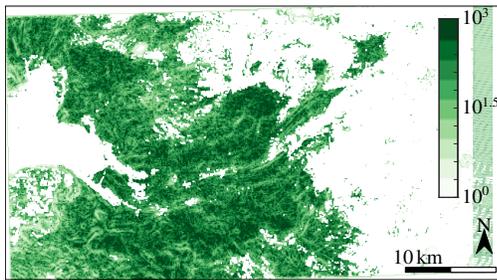
Corrected.

*- Line 247 and Figure 11: Actually, to my eye, it seems like the signal-to-noise ratio and major axis have a reasonable correlation, e.g. high SNR is inversely proportional to major axis. This would make sense as this means the cross-correlation peak is more concentrated and compact relative to the noise floor.*

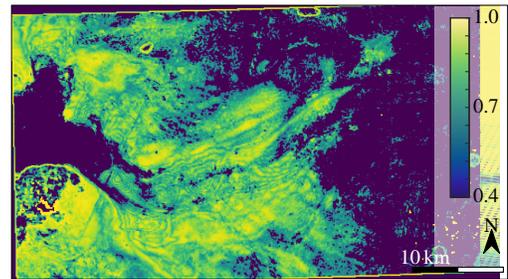
We agree, but this is only true for the extreme cases, the bulk of the matching is lumped together.

*Again, a map of total uncertainty (or even just major axis) would be illuminating when compared to the maps of SNR and correlation coefficient.*

The suggested metrics are shown in figure 6. Though the relation suggested by the reviewer can not be seen by us.



(a) peak to noise of the correlation score.



(b) absolute value of the maximum score in the correlation surface.

Figure 6: Score values for the Greenland case.

- *Line 257: How do frequency domain methods prescribe displacement at integer resolutions? It's well known that a real-valued shift between two signals in the time/spatial domain will lead to a ramp in the frequency domain. This ramp can be estimated to achieve sub-pixel resolution (Leprince et al., 2007).*

Here as well as mentioned in the study, correlation is calculated in the Fourier space, and transformed back to the spatial domain. Hence, a correlation surface is the result. Since the inverse Fourier transformation is a discrete form, smearing and peak-locking is happening.

In a first step (Leprince et al. 2007) do this procedure, to get the integer displacement. Then the window gets adjusted/refined to the displaced location, where again correlation is done in the Fourier-space. However, now there is no transformation back to the spatial domain, and a plane is fitted. The fitting of a plane/ramp overcomes the drawbacks of the inverse discrete fourier transformation.

- *Line 275: It would probably be useful to specify "physical signal" if one is referring to improvement of data assimilation/inverse methods.*

Included.